UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/674,364	09/29/2003	Michael Y. Lai	ITL.1482US (P16116)	5221

21906 7590 04/15/2008
TROP PRUNER & HU, PC
1616 S. VOSS ROAD, SUITE 750
HOUSTON, TX 77057-2631

EXAMINER	
TECKLU, ISAAC TUKU	

ART UNIT	PAPER NUMBER
2192	

MAIL DATE	DELIVERY MODE
04/15/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/674,364
Filing Date: September 29, 2003
Appellant(s): LAI, MICHAEL Y.

____
Timothy N. Trop, Reg. No. 28,994
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 01/22/2008 appealing from the Office action mailed.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix of the brief is correct.

(8) Evidence Relied Upon

US 6,226,789	Tye et al.	01/29/1996

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

- Claims 1-57 are rejected under 35 U.S.C. 102(b) as being anticipated by Tye et al. (US 6,226,789 B1).

As per claim 1, Tye discloses a method comprising:

generating an intermediate representation (IR) of a source program (e.g. FIG. 65B, element 884 "SOURCE INSTRUCTIONS " and related text), where the source program includes one or more instructions for processing data in a bit field within a data structure (col. 63, 15-20 "… builds an Intermediate representation (IR) …");

modifying the intermediate representation to more efficiently execute the one or more instructions for processing the bit field data (col. 63, 15-30 "… then modifies IR…"); and

generating resultant code based on the modified intermediate representation (col. 63, 15-35 "… resulting instruction … binary image …" and e.g. FIG. 43, CALL_FLAG 538 and related text).

As per claim 2, Tye discloses the method of claim 1, wherein modifying the intermediate representation further comprises: pre-processing the IR to perform preliminary modification of the IR (col. 2, 5-15 "… preprocessor to convert …") and e.g. FIG. 62C and related text).

As per claim 3, Tye discloses the method of claim 2, wherein modifying performing pre-processing further comprises:

performing data flow analysis to gather information regarding definition and usage of the bit field data (col. 1, 55-60 "… data flow analysis …" and e.g. FIG. 57 and related text.); and generating a def/use graph to classify the information (col. 1, 55-60 "… graph representation ...").

As per claim 4, Tye discloses the method of claim 3, wherein generating a graph further comprises:

generating a graph to classify the information in relation to an associated packet (col. 3, 55-60 "… data flow analysis …").

As per claim 5, Tye discloses the method of claim 2, wherein modifying the intermediate representation further comprises: (a) allocating a temporary variable to hold the bit field data (e.g. FIG. 23 and related text); and (b) modifying the IR so that the temporary variable is processed in accordance with the instructions (col. 63, 15-30 " … then modifies …").

As per claim 6, Tye discloses the method of claim 5, further comprising: (c) assigning the value of the temporary variable to a memory (e.g. FIG. 47 and related text).

As per claim 7, Tye discloses the method of claim 6, further comprising: performing steps (a), (b) and (c) for a single basic block (e.g. FIG. 47 and related text).

As per claim 8, Tye discloses the method of claim 7, further comprising: identifying two or more sub-blocks within the basic block (col. 65, 20-30 "… basic bloc …" and e.g. FIG. 47 and 48 and related text).

As per claim 9, Tye discloses the method of claim 8, wherein: steps (a), (b) and (c) are performed for each sub-block (e.g. FIG. 47 and related text).

As per claim 10, Tye discloses the method of claim 5, further comprising: determining whether all of the one or more instructions for processing the bit field data are read-after-write instructions; and performing steps (a) and (b) only if the determination is false (col. 47, 40-45 "… read write …").

As per claim 11, Tye discloses the method of claim 6, further comprising: determining whether any of the one or more instructions for processing the bit field data are write

instructions; and performing step (c) only if the determination is true (e.g. FIG. 49 and related text).

As per claim 12, Tye discloses the method of claim 6, further comprising: removing the modifications effected by steps (a), (b) and (c) upon determining that such removal is expected to provide an efficiency benefit in the resultant code (e.g. FIG. 62A, element 836 and related text).

As per claim 13, Tye discloses the method of claim 2, wherein pre-processing further comprises: disambiguating a memory reference to the bit field (e.g. FIG. 62B, element 844 and related text).

As per claim 14, Tye discloses the method of claim 1, wherein modifying the intermediate representation further comprises: modifying the IR so that multiple instructions to initialize respective bit fields of a data structure are performed with a single write to a memory (col. 63, 15-30 "… then modifies IR…").

As per claim 15, Tye discloses the method of claim 14, wherein the multiple instructions occur within a pre-defined maximal scope (e.g. FIG. 62B and related text).

As per claim 16, Tye discloses the method of claim 1, wherein modifying the intermediate representation further comprises: modifying the IR so that multiple read instructions for respective bit fields of a data structure are performed with a single read from a memory (col. 63, 15-30 "… then modifies IR…").

As per claim 17, Tye discloses the method of claim 16, wherein the multiple read instructions occur within a pre-defined maximal scope (e.g. FIG. 62B and related text).

As per claim 18, Tye discloses the method of claim 1, wherein modifying the intermediate representation further comprises: modifying the IR so that multiple write instructions to respective bit fields of a data structure are performed with a single write to

a memory (col. 63, 15-30 "… then modifies IR…").

As per claim 19, Tye discloses the method of claim 18, wherein the multiple read instructions occur within a pre-defined maximal scope (e.g. FIG. 62B and related text).

As per claim 20, Tye discloses the method of claim 1, wherein modifying the intermediate representation further comprises:

determining that a first instruction, being one of the one or more instructions, indicates a bit-wise logical operation on the bit field data (e.g. FIG. 65A, element 874 and related text);

determining that a second instruction of the source program indicates a bit-wise logical operation on a second bit field within the data structure (e.g. FIG. 65A, element 876 and related text); and

modifying the IR so that the first and second instructions are performed via a single read from a memory (e.g. FIG. 65A, element 880 and related text).

As per claim 21, Tye discloses the method of claim 20, wherein the bit-wise logical operation is a bit-wise OR operation (e.g. FIG. 42 and related text).

As per claim 22, Tye discloses the method of claim 20, wherein the bit-wise logical operation is a bit-wise AND operation (e.g. FIG. 42A and related text).

As per claim 23, this is the article version of the claimed method discussed above (Claim 1), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 24, this is the article version of the claimed method discussed above (Claim 2), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 25, this is the article version of the claimed method discussed above (Claim 3), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 26, this is the article version of the claimed method discussed above (Claim 4), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 27, this is the article version of the claimed method discussed above (Claim 5), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 28, this is the article version of the claimed method discussed above (Claim 6), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 29, this is the article version of the claimed method discussed above (Claim 7), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 30, this is the article version of the claimed method discussed above (Claim 8), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 31, this is the article version of the claimed method discussed above (Claim 9), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 32, this is the article version of the claimed method discussed above (Claim 10), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 33, this is the article version of the claimed method discussed above (Claim 11), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 34, this is the article version of the claimed method discussed above (Claim 12), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 35, this is the article version of the claimed method discussed above (Claim 13), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 36, this is the article version of the claimed method discussed above (Claim 14), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 37, this is the article version of the claimed method discussed above (Claim 15), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 38, this is the article version of the claimed method discussed above (Claim 16), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 39, this is the article version of the claimed method discussed above (Claim 17), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 40, this is the article version of the claimed method discussed above (Claim 18), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 41, this is the article version of the claimed method discussed above (Claim 19), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 42, this is the article version of the claimed method discussed above (Claim 20), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 43, this is the article version of the claimed method discussed above (Claim 21), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 44, this is the article version of the claimed method discussed above (Claim 22), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 45, this is the compiler version of the claimed method discussed above (Claim 1), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 46, this is the compiler version of the claimed method discussed above (Claim 2), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 47, this is the compiler version of the claimed method discussed above (Claim 3), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 48, this is the compiler version of the claimed method discussed above (Claim 5), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 49, this is the compiler version of the claimed method discussed above (Claim 6), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 50, this is the compiler version of the claimed method discussed above (Claim 14), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 51, this is the compiler version of the claimed method discussed above (Claim 15), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 52, this is the compiler version of the claimed method discussed above (Claim 17), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 53, this is the compiler version of the claimed method discussed above (Claim 19), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 54, this is the compiler version of the claimed method discussed above (Claim 20), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 55, this is the compiler version of the claimed method discussed above (Claim 21), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 56, this is the compiler version of the claimed method discussed above (Claim 21), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

As per claim 57, this is the compiler version of the claimed method discussed above (Claim 21), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Tye.

(10) Response to Argument

Appellant has argued as follows:

a) "The final office action suggests that bit fields are taught and that somehow the source program includes instructions for processing data in the bit field. If we accept that for argument purpose, we are left with the question – where is the modifying that intermediate representation to more efficiently execute the one or more instructions for processing the bit field data?" (Brief, page 11).

Examiner's Response:

As an initial matter, this issue has been addressed in an Advisory Action mailed on 11/06/2007, but Appellant has yet to acknowledge and/or has remained silent.

Examiner disagrees with Appellant's assertion. In response to applicant's argument that the intermediate representation is modified to more efficiently execute the one or more instructions for processing the bit field data, Tye discloses optimization technique such as data flow analysis, used in binary translation to handle any form of the IR (Intermediate Representation). The optimizer processes the list of code cells 600 of FIG. 45 to perform optimizations using a binary image as input. Generally, <u>optimizations reduce execution time and reduce system resource requirements of a machine executable program</u> (col. 64:1-30). Furthermore, a bit field is a common idiom used in computer programming to store a set of Boolean data type <u>flags</u> compactly, as a series of bits (emphasis added), Tye illustrates processing bit field in FIG. 43. Here at col. 57, lines 25-54, states "two types of example entries in the profile statistics 17c used to determine translation units of a routine are shown. The first entry type is a TARGET ADDRESS TYPE ENTRY 532 comprising a NON_NATIVE_TARGET_ADDRESS tag 536, a CALL_FLAG 538 and a COUNT 540. Each entry of this type comprises a unique non-native address 536 which is the target of a transfer of a control. A list of these entries is used to represent all the locations to which control has been transferred at run-time as recorded by the run-time interpreter in the profile statistics. <u>Each entry is unique from every other entry of the list</u>. The NON_NATIVE_TARGET_ADDRESS 536 functions as an identification tag or search index when searching for an entry amongst the profile statistics, as previously described, for example when the profile statistics are organized in a hash

table. The <u>CALL_FLAG 538 is a Boolean flag set to TRUE</u> (bit) when the associated NON_NATIVE_TARGET_ADDRESS has been the target of a routine CALL. Otherwise, CALL_FLAG is FALSE (bit), COUNT 540 is an integer representing the total number of times control has been transferred to the associated NON_NATIVE_TARGET_ADDRESS. For example, if an instruction set comprises four instructions that transfer control, COUNT represents the number of times the associated NON_NATIVE_TARGET_ADDRESS has been the target address to which control has been transferred by the four instructions" (emphasis added). That is to say the bit field here has been used to facilitate execution and/or control of such a CALL routine instruction efficiently when the bit field is True and when it is False.

Appellant has argued as follows:

b) "The asserted support for this limitation consists entirely of the clause '… then modifies the IR to produce a final version of the IR that corresponds to instructions in the second instruction set.' But this says nothing about the bit field data whatsoever. Not only does it even mention the bit field data in particular, but it in no way suggests that the intermediate representation is modified "to more efficiently execute the one or more instructions for processing the bit field data." Instead, it is indicated that the intermediate representation is modified to produce a final version" that corresponds to instructions in the second instruction set." (Brief, page 11).

Examiner's Response:

Appellant acknowledges that in Tye "the intermediate representation is modified to produce a final version that corresponds to instruction in the second instruction set" (emphasis added), but left us such a question as -- Is Tye's modified intermediate representation to produce final version with less efficient to execute instructions for processing bit field data in the second (different) instruction set (emphasis added)? Examiner would like to reiterate that Tye certainly modifies the intermediate representation to produce efficient code. Furthermore, Tye discloses "after constructing an initial IR as in step 810, the initial IR is translated and optimized to produce a final routine IR as in step 812 (emphasis added).

Furthermore, in response to applicant's argument that intermediate representation is modified to more efficiently execute the one or more instructions for processing the bit field data, a recitation of the intended use of the claimed invention (" to more efficiently execute the one or more instruction for processing the bit field data") must result in a structural difference between the claimed invention and the prior art in order to patentably distinguish the claimed invention from the prior art. If the prior art structure is capable of performing the intended use, then it meets the claim. That is to say, one has to prove that the prior art structure is incapable of efficiently executing the one or more instruction for processing the bit field data. See MPEP 2111.01.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Isaac T Tecklu/

Examiner, Art Unit 2192

Conferee

Tuan Q. Dam, Supervisory Patent Examiner, AU 2192

/Tuan Q. Dam/

Supervisory Patent Examiner, Art Unit 2192

Wei Zhen, Supervisory Patent Examiner, AU 2191

/Wei Zhen/

Supervisory Patent Examiner, Art Unit 2191